Untersuchung der Netzlastrobustheit von OPC UA - Standard, Profile, Geräte und Testmethoden

Sergej Gamper[1], Bal Krishna Poudel[1], Sebastian Schriegel[1], Florian Pethig[1], Jürgen Jasperneite[1,2]

[1]Fraunhofer IOSB-INA
Langenbruch 6, 32657 Lemgo
{sergej.gamper, bal.poudel, sebastian.schriegel, florian.pethig, juergen.jasperneite}@iosb-ina.fraunhofer.de

[2]Institut für Industrielle Informationstechnik inIT
Langenbruch 6, 32657 Lemgo
{juergen.jasperneite}@hs-owl.de

Abstrakt: Momentan werden neue Kommunikationstechnologien wie z. B. OPC UA, TSN oder auch SDN ausgearbeitet und eingeführt, um den Anforderungen einer Industrie 4.0 gerecht zu werden. Dabei konzentriert man sich verständlicherweise hauptsächlich auf die funktionalen Aspekte der jeweiligen Technologie. Dabei können bestimmte grundlegende Qualitätsmerkmale aber leicht unterschätzt oder einfach übersehen werden. Dieses Dokument beschäftigt sich mit der Fragestellung, ob das Thema der Netzlastrobustheit bei OPC UA im Standard oder bei der Zertifizierung der OPC UA-fähigen Geräte bereits eine Rolle spielt. Wie die Erfahrungen der letzten Jahrzehnte in der Nutzung der Ethernet-basierten industriellen Netzwerke (z. B. PROFINET) zeigen, spielt dieses Thema eine sehr wichtige Rolle.

1 Einleitung

Durch das Fortschreiten der vierten industriellen Revolution (Industrie 4.0) und den damit verbundenen zusätzlichen Anwendungen öffnet sich die industrielle Kommunikation immer mehr für die globale Vernetzung und die Dienste des Internets. Das Internet ist damit auch für die unterste Feldbusebene der industriellen Anlagen erreichbar und ermöglicht somit die weltweite Kommunikation industrieller Anlagen und Geräte untereinander [1]. Diese Art der Kommunikation unter Geräten wird im Kontext der Industrie 4.0 als „Industrial Internet of Things“ (IIoT) bezeichnet [2]. Mit dem Einzug des Internets in die industrielle Kommunikation kommen zusätzliche Kommunikationsstandards zum Einsatz, die bisher hauptsächlich im Office-Bereich angewandt worden sind, wo die QoS-Anforderungen wie Verfügbarkeit, im Vergleich zur industriellen Anwendung, deutlich niedriger sind. Einer der aktuell am stärksten vorangetriebenen M2M-Kommunikationsstandards in der Industrie ist OPC UA [3].

OPC UA soll in der Zukunft durch Weiterentwicklungen des Standards (z. B. in Richtung zyklischer Kommunikation mittels Publisher Subscriber (PubSub) Pattern) immer tiefer, d.h. bis in die Sensor- und Aktor-Ebene in Fabriken und industriellen Anlagen vordringen. In Kombination mit Ethernet-TSN [4] sollen auch hoch performante Anwendungen, wie z. B. „Motion Control“, damit realisierbar werden [5]. In dieser Umgebung hat das Thema Verfügbarkeit und als Folge die Netzlastrobustheit sowohl der Kommunikationsnetzwerke als auch der einzelnen Kommunikationselemente eine zentrale Bedeutung. Man denke dabei an Szenarien bei denen eine Produktionsanlage oder ein Teil davon ausfällt, weil das Kommunikationsnetzwerk oder Empfangsressourcen bei einem einzelnen ressourcenbeschränkten Netzwerkgerät kurzzeitig überlastet wurden. Dies könnte z. B. durch Übertragen einer größeren Datenmenge oder durch einen gezielten Angriff (DoS - Denial of Service), oder auch z. B. durch falsche Netzwerkkonfiguration oder durch Fehlfunktion eines anderen Gerätes im Netzwerk auftreten. Ungeplante Betriebsunterbrechungen verursachen in der Regel hohe Verluste für den Anlagenbetreiber. Der Überlastschutz eines Netzwerks kann mit Hilfe von TSN gewährleistet werden [6], die einzelnen Geräte müssen aber trotzdem robust gegen eine mögliche Überlast sein, da die Nutzung von TSN optional ist und nicht immer vorausgesetzt werden kann. Aus diesem Grund soll das Thema der Netzlastrobustheit von OPC UA-fähigen Geräten unabhängig von TSN in diesem Paper genauer adressiert werden.

2 Stand der Technik OPC UA

2.1 OPC UA Kommunikationsmodell

OPC UA bietet folgende zwei Mechanismen zum Austausch von Informationen in verschiedenen Use Cases: Client-Server-Modell und Publisher-Subscriber-Modell. Im Client-Server-Modell muss ein OPC-UA-Client einen dedizierten Kommunikationskanal mit dem Server aufbauen, um auf die Daten oder Funktionen des Servers

zugreifen zu können. Die Anzahl der gleichzeitigen Verbindungen wird oft durch die Speicher- und Rechen-Ressourcen des Servers beschränkt, da die Verbindungsinformationen aller verbundenen Clients im Speicher gehalten werden. Im Fall einer verschlüsselten Kommunikation entsteht eine zusätzliche Prozessorlast. Des Weiteren wird die Prozessorlast auch durch unterschiedliche Aktualisierungsraten der von Clients angeforderten Informationen erhöht [7]. Im Publisher-Subscriber-Modell (PubSub) muss es dagegen keinen dedizierten bzw. direkten Kommunikationskanal zwischen Publisher und Subscriber geben. Das Routing der Nachrichten bzw. der einzelnen Informationen wird in diesem Fall von der nachrichtenorientierten Middleware (Message Oriented Middleware - MOM) übernommen. Dieser Ansatz führt zu einem niedrigeren Ressourcenverbrauch auf der Publisher-Seite im Vergleich zur direkten gleichzeitigen Client-Server-Kommunikation mit mehreren Clients. Ein OPC UA-Publisher sendet dabei eine vorkonfigurierte Teilmenge der Daten (Data Set) an die MOM mit einer vorkonfigurierten Zyklusrate und der OPC UA-Subscriber empfängt dann die Netzwerknachricht nach Bedarf von der MOM. MOM kann in zwei verschiedenen Varianten implementiert werden: Broker-basiert oder brokerlos. Die brokerlose Variante kann mit dem UADP (UDP-based UA) Protokoll in Form von UDP Multicasts und unter Verwendung der Netzwerkinfrastruktur realisiert werden. Eine brokerbasierte MOM kann nach aktuellem Stand auf den Protokollen AMQP oder MQTT basieren [8].

2.2 OPC UA Transportprotokolle

OPC UA verwendet zwei TCP/IP basierte Protokolle für die Kommunikation zwischen Client und Server. Das sind Binärprotokoll und Webservice. Es kann auch eine Kombination (Hybrid) der beiden Protokolle verwendet werden, die im Wesentlichen die Vorteile beider Protokolle ausnutzt.

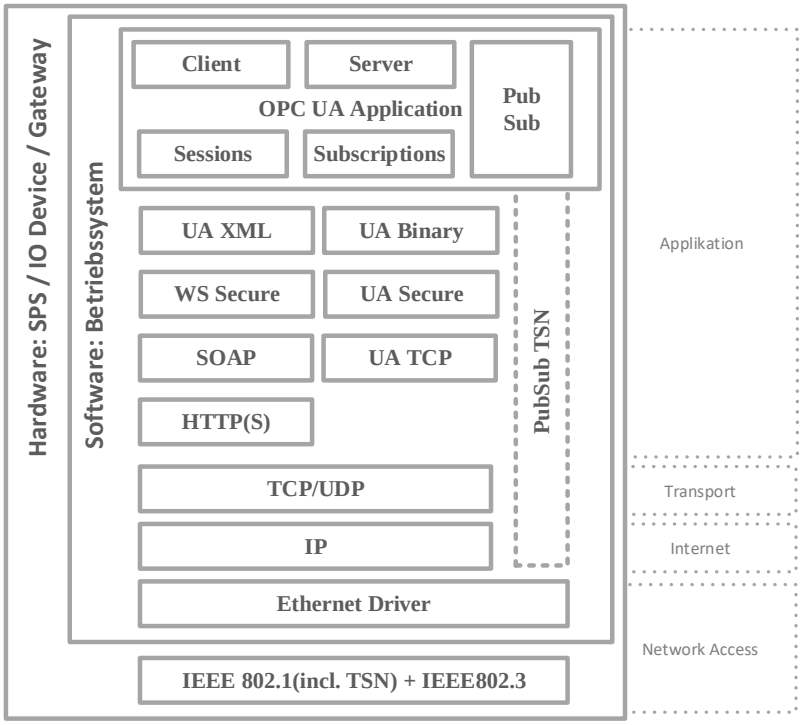

Abbildung 1 Stackarchitektur von OPC UA in einer Gerätearchitektur

Die Eigenschaften der beiden Protokolle lassen sich wie folgt darstellen.

- **Binärprotokoll (UA Binary)** ist sehr Ressourcen- und Overhead-effizient, erreicht deshalb eine gute Performanz und ist insbesondere für Ressourcen-beschränkte eingebettete Systeme gut geeignet. Durch präzise Spezifikation lässt das Protokoll nur wenig Freiheitsgrade bei der Verwendung zu, was sich wiederum positiv auf die Interoperabilität der unterschiedlichen Lösungen auswirkt. Das Binärprotokoll ist für alle OPC UA-Geräte vorgeschrieben und benutzt einen festen TCP Port (4840)
- **Webservices (XML-SOAP)** ist ein optionales Transportprotokoll, welches durch Verwendung von „Simple Object Access Protocol“ (SOAP) flexibler bei der Datenrepräsentation als das Binärprotokoll ist, verursacht aber mehr Overhead durch XML-kodierte Nachrichten und ist aus diesem Grund langsamer. Das Protokoll lässt den Einsatz von .NET und Java Komponenten zu, findet aber wegen einem

höheren Ressourcen-Bedarf kaum Einsatz in eingebetteten Geräten. Durch die Verwendung des HTTPS-TCP-Ports (443) können auch Implementierungen hinter Firewalls erreicht werden.

Die Verwendung einer Kombination aus beiden Protokollen (UA-Binary über HTTPS) ist optional und basiert im Wesentlichen auf der Binärkodierung, die über Port 443 transportiert wird. Somit werden die wesentlichen Vorteile der beiden Protokolle ausgenutzt. D.h. das Protokoll kombiniert eine gute Performanz mit einer guten Firewall-Kompatibilität.

Zusammen mit TCP/IP kommt noch eine Reihe von Standard-Hilfs-Protokollen in den OPC UA-Geräten zum Einsatz, wie z. B. das „Address Resolution Protocol“ (ARP) und das „Dynamic Host Configuration Protocol“ (DHCP), die das Gerät u.U. anfälliger für bestimmte Netzwerklast-Arten machen können.

2.3 OPC UA Profile

OPC UA ist ein mehrteiliger Standard, der die Anzahl der in der IEC 62541-4 spezifizierten Dienste und in der IEC 62541-5 eine Vielzahl von Informationsmodellen beschreibt. Diese Dienste und die Informationsmodelle werden im Rahmen der OPC UA-Spezifikation als Features bezeichnet. Die Gruppierung von Features, die als eine Einheit getestet werden können, wird als "Conformance Unit" bezeichnet [9]. Profile sind in Teil 7 der OPC UA-Spezifikation definiert und aggregieren die vordefinierten Conformance Units und andere Profile. Eine OPC UA-Anwendung (Client oder Server) sollte alle obligatorischen Conformance Units und Profile enthalten, um mit einem bestimmten Profil kompatibel zu sein. OPC UA-Profile werden hauptsächlich für den Test und die Zertifizierung der OPC UA-Produkte definiert und beschreiben die Funktionalität, die eine bestimmte OPC UA-Anwendung bietet. Das Testen eines Profils besteht aus dem Test aller einzelnen Conformance Units, aus denen ein Profil besteht. OPC UA-Profile sind in vier Kategorien aufgeteilt Client, Server, Sicherheit und Transport. Die Client-Profilkategorie spezifiziert die Funktionen eines OPC UA-Clients. Entsprechend legt die Server-Profilkategorie die Funktionen eines OPC-UA-Servers fest, usw. Ein Profil gilt als vollwertig, wenn es alle Funktionen unterstützt, die für den Aufbau einer minimalen funktionalen OPC UA-Anwendung notwendig sind. Profile machen OPC UA skalierbar und erlauben den Einsatz von OPC UA sowohl in ressourcenbeschränkten Embedded-Anwendungen als auch in High-Level Enterprise-Systemen [10]. Folgende Profile von OPC UA wurden bis jetzt in IEC 62541-7 für Server definiert [9]:

- **Nano Embedded Device 2017 Serverprofil** - beinhaltet den Core OPC UA-Server Facet und das OPC UA TCP Binary Protokoll als erforderliches Transportprofil. Es eignet sich für chipbasierte Geräte mit begrenzten Ressourcen und kann für die industrielle Automatisierung eingesetzt werden.
- **Micro Embedded Device 2017 Serverprofil -** unterstützt alle Funktionen, die im Serverprofil des Nano Embedded Device enthalten sind. Darüber hinaus unterstützt es die Unterstützung von Subscriptions über die Embedded Data Change Subscription Server Facette und die Unterstützung von mindestens zwei Sitzungen.
- **Embedded 2017 UA-Server-Profil -** unterstützt alle Funktionen, die im Serverprofil des Micro Embedded Device enthalten sind. Darüber hinaus unterstützt es die Sicherheit durch die Sicherheitsrichtlinien und die Unterstützung der Standard DataChange Subscription Server Facette. Es ist für den Einsatz mit Embedded-Geräten mit mindestens 50 MByte Speicher und leistungsfähigerem Prozessor vorgesehen.
- **Standard 2017 UA-Serverprofil 2017 -** basiert auf dem Embedded UA Server Profil und definiert minimale Funktionalitäten für PC-basierte OPC UA-Server und unterstützt die Discovery Services. Darüber hinaus erfordert es höhere Anzahl der gleichzeitigen Sitzungen, Subscriptions und überwachten Elemente im Vergleich zu eingebetteten Profilen.
- **Global Discovery Server-Profil** - dabei handelt es sich um ein „Full Featured“-Profil, das die notwendigen Dienste und das Informationsmodell eines UA-Servers abdeckt, der als Global Discovery Server(GDS) fungiert.
- **Global Discovery und Zertifikatsmanagementserver** - ist ein „Full Featured“-Profil, das die notwendigen Dienste und das Informationsmodell eines UA-Servers abdeckt, der als GDS und globaler Zertifikatsmanager fungiert.

2.4 OPC UA Security Model

Die Sicherheit von OPC UA basiert auf Mechanismen wie Authentifizierung, Autorisierung und Verschlüsselung. Es stehen drei Sicherheitsoptionen/-stufen zur Verfügung: "None", "Sign" und "SignAndEncrypt" [11]. Ein OPC UA Server gibt die von ihm unterstützten Sicherheitsmechanismen vor. Bei mehreren Möglichkeiten entscheidet der Client je nach Bedarf welche von dem Server unterstützten Sicherheitsmechanismen bei der Kommunikation eingesetzt werden. Wie die Sicherheitsanalyse von OPC UA durch das Bundesamt für Sicherheit in der Informationstechnik (BSI) gezeigt hat, bietet die Sicherheitsoption „None“ wenig bis keinen Schutz vor IT-Angriffen. Ein hohes Maß an Sicherheit bietet dagegen die Option SignAndEncrypt [12]. Da in dem industriellen Umfeld und besonders auf der Feld-/Maschinen-Ebene aber hauptsächlich Ressourcen-limitierte eingebettete Systeme eingesetzt werden und gleichzeitig hohe Anforderungen an die Kommunikationsleistung gestellt werden, wird oft die Sicherheitsoption "None" oder „Sign“ verwendet. Der Einsatz der Verschlüsselung (SignAndEncrypt) findet hauptsächlich beim Transport der sensiblen Daten über Domaingrenzen hinweg statt. Als Domain wird in diesem Zusammenhang je nach Sicherheitsanforderung das Netzwerk auf der Maschienen-, Anlagen- oder Fabrik-Ebene gemeint.

2.5 OPC UA Certification Test

Der Kommunikationsstandard OPC UA wurde mit dem Ziel entwickelt, Interoperabilität zwischen Produkten unterschiedlicher Anbieter zu ermöglichen. Derzeit gibt es verschiedene OPC UA-Anwendungen, die SDKs von verschiedenen Anbietern verwenden. In diesem Szenario ist es notwendig, durch Zertifizierung die Kompatibilität und Korrektheit gegenüber dem OPC UA Standard für die Anwender zu gewährleisten. Die Zertifizierung eines Produkts gibt den Anwendern Sicherheit bei der Auswahl zertifizierter Produkte.

Das Zertifizierungs- und Compliance-Programm der OPC Foundation bietet den Anbietern der OPC UA-Lösungen die Möglichkeit, ihre Produkte zu testen, um Compliance, Interoperabilität, Robustheit, Usability und Ressourceneffizienz sicherzustellen [13]. Das OPC UA Compliance Test Tool (UACTT) bewertet die Konformität der OPC UA-Produkte mit der Spezifikation. Es testet alle obligatorischen sowie die optionalen Funktions- und Conformance-Units, die vom Produkt unterstützt werden. Es beinhaltet auch Testfälle, ob ein Server die ungültige Client-Anfrage bearbeiten und beantworten kann und umgekehrt. Im Interoperabilitätstest werden OPC UA-Produkte mindestens mit fünf verschiedenen OPC UA-Produkten von unterschiedlichen Herstellern getestet. In einem Robustheitstest wird im Wesentlichen die Fähigkeit des Produkts zur Handhabung und Wiederherstellung der verlorenen Kommunikation überprüft. In einem Effizienztest wird das Produkt 36 Stunden lang unter „Last" getestet. So kann beispielsweise ein zu testender Server aus 1000 dynamisch wechselnden Knoten bestehen, die von mindestens fünf verschiedenen Clients gleichzeitig abonniert werden. In diesem Zusammenhang wird die CPU- und RAM-Nutzung durch den OPC UA-Server untersucht. In einem Usability-Test wird die Genauigkeit der Dokumentation des Produkts getestet und es wird überprüft, ob sich das Produkt wie vorgesehen verhält. Das Testtool enthält Testszenarien sowohl für Client- als auch für Server-Applikationen. Dabei werden einzelne Profile, Facets und Conformance Units getestet. Sobald das Produkt alle diese Tests erfolgreich bestanden hat, kann die Zertifizierung des Produkts beantragt werden.

Derzeit fokussieren sich die Zertifizierungstests für OPC UA-Lösungen hauptsächlich auf die im Standard vorgeschriebenen funktionalen Aspekte, um eine bessere Interoperabilität der Lösungen zu garantieren. Die Robustheit der Endgeräte gegenüber der Netzlast wird aber kaum berücksichtigt. Dabei spielt das Thema im industriellen Umfeld eine sehr bedeutende Rolle. Insbesondere wenn auch die kleinsten Geräte, wie z. B. einzelne Sensorknoten per OPC UA erreichbar werden sollen.

Die Geschichte der Ethernet-basierten industriellen Kommunikationssysteme, zeigt, dass die Robustheit der einzelnen Netzwerkkomponenten gegen Netzbelastung in einem Automatisierungsnetzwerk eine große Herausforderung ist. Dafür wurden innerhalb mancher Kommunikationsstandards spezielle Mechanismen entwickelt, um die Bandbreite für die übertragenen Prozessdaten zu sichern (wie z. B. bei PROFINET-IRT). Diese Mechanismen erfordern den Einsatz von speziellen Netzwerkkomponenten. Trotzdem konnte das Problem der Netzlast nicht vollständig gelöst werden, da es außer Prozessdaten auch andere Dienste gibt, die auf Standard-Protokollen wie UPD/IP oder SNMP basieren und die ein unverzichtbarer Bestandteil des Kommunikations-Standards sind. Aus diesem Grund wurden bei PROFINET die Anforderungen an die Geräte bezüglich Netzlast-Robustheit genau definiert und deren Einhaltung im Zertifizierungs-Test verankert. Im folgenden Kapitel werden

die wesentlichen Netlast-Tests, die im Zuge der Geräte-Zertifizierung bei PROFINET durchgeführt werden, beschrieben.

3 Stand der Technik Security Level 1 Test bei PROFINET

Bei PROFINET ist das Thema der Netzlastrobustheit gut definiert und die entsprechenden Tests sind fester Bestandteil der Gerätezertifizierung. Bei der Zertifizierung der PROFINET-IO-Device werden neben den funktionalen Tests spezifizierte Netzlasttests durchgeführt, bei denen das Verhalten der DUTs (Device Under Test) unter genau definierten Bedienungen geprüft wird. Der Beispielaufbau des Tests wird in der folgenden Abbildung 2 dargestellt.

Während des Tests wird eine aktive IO-Beziehung zwischen einem IO-Controller und dem DUT aufgebaut, wobei sowohl zyklische als auch azyklische Daten zwischen dem IO-Controller und dem DUT ausgetauscht werden. Auf diese Weise wird die Verfügbarkeit der Kommunikations-Funktionen des DUTs während des Tests kontrolliert. Die Testframes werden von dem Last-Generator erzeugt und entweder über den Switch oder bei Verfügbarkeit über den zweiten Port des DUTs ins Netzwerk eingespeist.

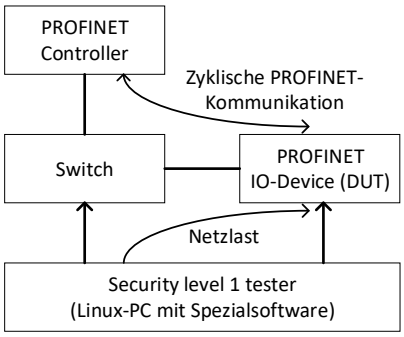

Abbildung 2: Testaufbau bei 2-Port DUT

Entsprechend der beim Test erreichten Robustheit gegen die Netzlast wird dem DUT eine der drei im Standard definierten Robustheits-Klassen zugeordnet, die sich in Grund-Robustheit (Klasse I), Standard-Robustheit (Klasse II) und Erweiterte-Robustheit (Klasse III) aufteilen. Für die erfolgreiche Zertifizierung muss das DUT mindestens die Klasse I erreicht haben.

Die Netzlasttests der Klasse I sind für kleinere Anlagen relevant, die bis zu 50 IO-Devices und 1-2 Controller enthalten können. Die Netzlasteinstellungen für die Tests der Klasse II gehen von einer größeren PROFINET-Domain mit bis zu 1000 Knoten aus, wodurch eine höhere Grundlast zu erwarten ist. Die Tests der Klasse III sind für PROFINET-Applikationen mit erhöhten Netzlastanforderungen relevant, die insbesondere in konvergenten Netzwerken zum Einsatz kommen, in denen aus Verfügbarkeits- und Sicherheits-Gründen mit einer hohen Netzwerkbelastung gerechnet werden muss. Die für den Test generierte tatsächliche Last hängt von den verwendeten Protokoll-Frames, Art des Tests (Unicast, Multicast, Mailformed) und von der Netzlastklasse ab. In den nachfolgenden Tabellen werden auszugsweise wesentliche Netzlastszenarien aus PROFINET-Zertifizierung genauer beschrieben. Für die Berechnung der Last wird folgende Formel verwendet:

$$\%Netload@ms = \frac{GrossData}{Bandwidth@ms} \quad [14]$$

Unter Gross-Data werden alle über die Leitung übertragene Bytes inklusive Inter Frame Gap (12 Bytes), Präambel (8 Bytes) und Ethernet-Trailer (4 Bytes) zusammengefasst.

Die beim Test generierte Netzlast wird in zwei Leistungsstufen unterteilt „Normale Funktion“ und „Überlast Kommunikation“.

- Normale Funktion – vom Testsystem wird eine moderate Belastung des Netzwerkes bzw. des DUT generiert, die eine normale Arbeitsbelastung des Netzwerkes simuliert. Unter dieser Last soll das DUT sowohl zyklische als auch azyklische Kommunikation sowie die Switching-Funktion bei Mehrportgeräten aufrechterhalten und sonst keine Art Störung zeigen [15]. Die Kommunikation mit dem IO-Controller darf nicht unterbrochen werden.
- Überlast-Communication – bei diesem Scenario wird das Netzwerk bzw. das DUT so stark belastet, dass dies zu Aussetzern oder einem vollständigen Abbruch der Kommunikation führen kann. Das DUT darf dabei die Switching-Funktion nicht unterbrechen und nicht neustarten. Nach Abklingen der Belastung

muss das Device seine Funktion wieder fortsetzen. D.h. das Device muss wieder erreichbar sein und im Stande sein, eine neue Kommunikationsbeziehung aufzubauen [15].

Die Netzlast besteht in beiden Fällen sowohl aus gerichteten als auch aus nicht gerichteten Frames. Gerichtete Frames sind alle Frames, die das DUT direkt über MAC- Adresse (Unicast, Multicast oder Broadcast) adressieren. Die Auswahl der bei der Lasterzeugung benutzten Frames richtet sich nach den bei PROFINET eingesetzten Kommunikations-Protokolle, wie IP/UDP, RPC, DCP, ARP, MRP, LLDP, SNMP. Die Tabelle 1 zeig einen nicht vollständigen Auszug der bei der PROFINET-Zertifizierung eingesetzten Lastszenarien, die auch für die Robustheits-Tests der OPC UA-Geräte relevant sein könnten.

Art der Netzwerklast	Protokol	Sub-Protokol	Klasse I	Klasse II	Klasse III	Überlast
Gerichtete Netzwerklast (Multicast (MC) und Broadcast(BC))	All (MC and BC)	All (except below	< 0,01%@1 ms	< 1%@1 ms	< 5%@1 ms	100%
	ARP (BC)	ARP.req ARP-Probe.req	<1Frame@1 ms	<1Frame@1 ms 50 undirected 950 ms pause	< 5%@1 ms 999 undirected 1 directed	
		ARP Announce	< 0,01%@1 ms	< 0,01%@1 ms	< 5%@1 ms	
	IP UDP	(BC)	< 0,01%@1 ms	< 5%@1ms	< 5%@1 ms	
	IPv6 All	(MC)	< 0,01%@1 ms	< 1%@1 ms	< 5%@1 ms	
Nicht gerichtete NRT Last	All (MC)	Filtered in Step 1	< 10%@1 ms	< 10%@1 ms	< 10%@1 ms	
Gerichtete NRT Last	All (MC and BC)	All (except below)	< 0,01%@1 ms	< 1%@1 ms	< 5%@1 ms	
	ARP (BC)	ARP.req ARP-Probe.req	<1Frame@1 ms 50 undirected 950 ms pausen	<1Frame@1 ms 999 undirected 1 directed	< 5%@1 ms	
		ARP Announce	< 0,01%@1 ms	< 0,01%@1 ms	< 5%@1 ms	
	IP	UDP (BC)	< 0,01%@1 ms	< 5%@1ms	< 5%@1 ms	
	IPv6	All (MC)	< 0,01%@1 ms	< 1%@1 ms	< 5%@1 ms	

Tabelle 1: Netzwerklast gerichtet und nicht gerichtet

Neben den Lasten, die durch reguläre Frames der relevanten Protokolle erzeugt werden, werden auch Szenarien mit verfälschten bzw. nicht gültigen Protokoll-Frames getestet. Bei diesen Tests werden zwischen den korrekten Frames einzeln (1 Frame pro 100 ms) verfälschte Protokoll-Frames eingespeist. Bei diesen Frames werden bewusst Protokoll-relevante Stellen, wie z. B. IP-Adressen, Parameter in IP-/UDP-Header oder Payload verfälscht bzw. manipuliert. Mit dem Ziel mögliche Schwachstellen in der Implementierung des eingesetzten Kommunikations-Stacks oder darauf basierenden Modulen zu identifizieren. Für alle drei Robustheits-Klassen gelten dabei gleiche Testbedienungen. In der Tabelle 2 werden die einzelnen Testszenarien mit der Art der Verfälschung der Protokoll-Frames dargestellt.

Art der Netzwerklast	Protocol	Testszenario	Klasse I/II/III Normale Funktion
Gerichtete Netzwerklast	IP	Valid Ethernet frame but malformed IP frames (UC)	1 frame @ 100 ms
		Frames where IP-Source Address and IP-Destination Address are identical (UC)	
		Frames where IP-Source Address is a Multicast- or Broadcast-Address (UC)	
	IP / ICMP	Valid Ethernet frame but malformed ICMP frames (UC)	
		Frames without payload (UC)	
		Frames with maximum payload (UC)	
	IP / UDP	Valid Ethernet frame but malformed UDP frames (UC)	
		Frames without payload (UC)	
		Frames with maximum payload (UC)	
	IP / UDP / SNMP / GET	Valid Ethernet frame but malformed SNMP / GET frames (UC)	
		Frames without payload UC)	
		Frames with maximum payload (UC)	
	IP / UDP / SNMP / SET	Valid Ethernet frame but malformed SNMP / SET frames (UC)	
		Frames without payload (UC)	
		Frames with maximum payload (UC)	

Tabelle 2: Last mit verfälschten Protokoll-Frames

Ja nach vom DUT unterstützte Verbindungsgeschwindigkeit generiert das Testsystem die Test-Last bezogen entweder auf 100Mbit/s oder 1000Mbit/s.

4 Exemplarischer Netzlasttest einer OPC UA Implementierung

Im Rahmen dieser Arbeit wurde exemplarisch mit einem OPC UA-Server der im Kapitel 3 beschriebene Test aus der PROFINET-Zertifizierung durchgeführt. In den nächsten Kapiteln werden das getestete OPC UA-Gerät die Testdurchführung und Testergebnisse beschrieben.

4.1 Beschreibung des zu testenden OPC UA Gerätes

Für den Test wurde ein kompakter Open-Source-Industrie-PC (Revolution PI der Firma KUNBUS) verwendet. Die Lösung basiert auf dem weit verbreiteten Einplatinencomputer Raspberry Pi und hat aufgrund des verwendeten SoC BCM2837 mit seiner Quad-Core ARM-CPU mit 1,2 GHz und 1 GByte RAM, ausreichend Leistung auch für komplexere industrielle Anwendungen. Mit dem mitgelieferten Betriebssystem (Raspbian) lassen sich benutzerspezifische Linux-basierte Anwendungen implementieren. Zum Testzweck wurde auf dieser Plattform ein exemplarischer OPC UA-Server implementiert. Dafür wurde die Open-Source-Bibliothek „open62541“ verwendet. Die Open-Source-Implementierung mit der Lizenz MPL v2.0 kann auch in kommerziellen Produkten eingesetzt werden. Aus diesem Grund ist es denkbar, dass die „open62541“-Bibliothek auch in den industriellen OPC UA-Lösungen einen breiten Einsatz findet.

Als Server-Applikation wurden zu Testzwecken zwei Methoden und zwei Variablen implementiert. Die Variablen beinhalten je einen Zählerwert, welcher pro Sekunde um eins erhöht wird. Dies dient der Kontrolle der Verbindung zwischen Server und Client und der Überwachung der Funktion des Servers, indem während des Tests die Änderung der Variablenwerte beobachtet wird.

Die realisierte Servervariante entspricht einem Micro Embedded Device Server Profil. Die Gesamtgröße der kompilierten Server-Binary betrug dabei ca. 1,3 MB. Aufgrund der guten Leistungseigenschaften der verwendeten Hardware wären auch komplexere Server-Profile mit höheren Leistungsanforderungen realisierbar.

4.2 Aufbau und Durchführung des Tests

In dem Testaufbau werden ähnlich, wie in der Abbildung 2 dargestellt, neben dem DUT und dem Netzlastgenerator noch ein industrietauglicher 4-Port Ethernet-Switch mit 100Mbit/s Datenübertragungsrate pro Port und ein PC-basierter OPC UA-Client (UAExpert) anstelle eines IO-Controllers eingesetzt. Da der DUT nur einen Ethernet-Port besitzt, wird die vom Lastgenerator erzeugte Test-Last über den Switch eingespeist. Während des Tests wird die Funktion des Servers mit Hilfe von UAExpert überwacht. Für diesen Zweck wurden in der Applikation des OPC UA-Servers zwei Zähler-Objekte eingebaut, die pro Sekunde ihren Zählerwert um 1 erhöhen. Wäre die Verbindung zwischen OPC UA-Server und Client während des Tests für längere Zeit unterbrochen, würden sich die abonnierten Zählerwerte im Client nicht mehr aktualisieren und ein Verbindungsabbruch würde im Logfenster des Clients gemeldet. Die Werte dieser Zähler werden zyklisch ca. alle 200 Millisekunden an UAExpert übertragen. Dies ist für dieses OPC UA-System (UAExpert und Server) die kleinsteinstellbare Aktualisierungszeit. Als erstes werden die Tests für die Netzlast-Klasse I „Normal Operation“ durchgeführt. Nach einem erfolgreichen Abschluss des vorherigen Tests wird der Test dann mit der nächst höheren Klasse II und dann entsprechend mit Klasse III wiederholt. Am Ende wird dann noch der Überlasttest durchgeführt um Robustheit des DUTs auf Überlastsituationen zu testen.

4.3 Testergebnisse und deren Auswertung

Alle Tests der Klassen I bis III wurden vom DUT bestanden. Es wurden keine Unterbrechungen in der Kommunikation zwischen dem OPC UA-Server und Client registriert. Unterbrechungen gab es erwartungsgemäß nur bei Überlasttests (z. B. bei ARP-Requests). Nach Abklingen der Überlastsituation konnte die OPC UA-Kommunikation aber problemlos neu aufgebaut werden.

Das scheinbar problemlose Bestehen der Last-Tests wurde durch folgende spezifische Umstände begünstigt:

- Die beim Test verwendete Aktualisierungszeit von 200 Millisekunden entspricht nicht den üblichen Zykluszeiten von 1-32 Millisekunden, die in den industriellen PROFINET-IO-Systemen hauptsächlich angewendet werden, wo z. B. bei einer Aktualisierungszeit von 1ms schon nach 3ms der Watchdog der Verbindungsüberwachung auslöst und die bestehende Verbindung abgebaut wird.
- Darüber hinaus bildet ein einfaches Client-/Server-Kommunikationsmodell nicht vollständig das Modell der üblichen IO-Kommunikation in den industriellen Feldbussystemen ab. Damit die Daten nicht nur, wie in unserem Fall, vom Server zum Client sondern auch zurück fließen können, wäre noch eine weitere und richtungsumgekehrte Server-/Client-Kommunikationsbeziehung notwendig.
- Die sehr einfach gehaltene Testapplikation und das ressourcensparende Profil der OPC UA-Anwendung ist eindeutig unterdimensioniert für die verwendete Hardwareplattform, wodurch dem Kommunikationsstack ausreichend Rechenressourcen zur Verfügung standen, um eine erhöhte Netzwerklast abzuarbeiten. In den realen industriellen eingebetteten Systemen ist dies aber aus Kostengründen selten der Fall.
- Die beim Test zur Lasterzeugung verwendeten Telegramme wurden speziell für PROFINET-Geräte konzipiert und erreichen u. U. in anderen Protokollstacks nicht die gewünschte Einsatztiefe und werden deshalb bereits nach der Prüfung des Ethernet-Headers verworfen.

Bemerkung: Zyklische Echtzeitkommunikation mit kleinen Übertragungszyklen ist kritisch für das Thema Netzlastrobustheit. Mit der Einführung von PubSub und TSN muss für OPC UA also das Thema Netzlastrobustheit explizit betrachtet werden.

5 Zusammenfassung

Es gibt aktuell viele sowohl kommerzielle als auch Open Source basierte Lösungen für OPC UA und TCP/IP Software-Stacks von unterschiedlichen Anbietern, die bereits Anwendung in industriellen Geräten finden. Es ist denkbar, dass es in vielen Gerätelösungen unterschiedlicher Hersteller auch viele Kombinationen aus den verfügbaren OPC UA und TCP/IP-Software-Stacks gibt. Es ist auch damit zu rechnen, dass diese Anzahl der verfügbaren Lösungen und Kombinationen davon in Zukunft noch stark zunehmen wird, da diese u.a. die Basis für die Industrie 4.0 darstellen. Auch die Weiterverbreitung von IPv6 bzw. deren Implementierungen in den

eingebetteten Systemen bringt neben neuen Möglichkeiten auch neue Risiken mit sich, die sich negativ auf die Netzlaststabilität der industriellen OPC UA-Applikationen auswirken könnten.

Alle in den offiziellen OPC-UA-Zertifizierungstests genannten Testkategorien basieren auf den Kernfunktionen der Server und Clients. In der industriellen Anwendungsdomäne können jedoch Hunderte verschiedene Geräte mit demselben Kommunikationsnetzwerk verbunden sein. Ein besonderes Thema dabei ist die Konvergenz der industriellen Protokolle, die in selben Netzwerk eingesetzt werden können. Viele dieser Protokolle benutzen auf eine oder andere Weise Multicast- und Broadcast-Telegramme, z. B. bei der Gerätesuche, die von vielen unbeteiligten Geräten im Netzwerk empfangen werden können. Darüber hinaus kann nicht verhindert werden, dass versehentlich oder mit böswilliger Absicht Geräte an das Netzwerk angeschlossen werden, die eine zusätzliche Netzwerklast erzeugen und so den Fluss des Netzwerkverkehrs stören können.

Wie der in Kapitel 4 beschriebene Test gezeigt hat, ist das klassische Server-Client Kommunikationsmodell nicht unbedingt anfällig für die verwendete Testnetzlast. Es ist aber auch aufgrund von zu großen Aktualisierungszeiten und Protokolloverhead nur bedingt für den industriellen Einsatz in der Feldebene geeignet. Durch den Einsatz des PubSub-Kommunikationsmodells erfüllt OPC UA zwar die höheren industriellen Kommunikationsanforderungen und es können damit viel kleinere Aktualisierungszeiten erreicht werden, gleichzeitig steigt aber die Kommunikationsbelastung der einzelnen Kommunikationsteilnehmer und des gesamten Netzwerks um ein Vielfaches an. Dieses Kommunikationsaufkommen entspricht dann etwa dem der klassischen Industrieprotokolle und reduziert die ohnehin schon oft begrenzte Rechenleistung der beteiligten Geräte. In dieser Situation spitzt sich die Frage der Netzlastrobustheit der einzelnen Geräte entsprechend zu. Wie verhalten sich Geräte unter diesen Umständen? Ist ein bestimmtes Verhalten korrekt oder unzulässig? Funktioniert die Anwendung weiterhin und bleiben alle etablierten Kommunikationskanäle zwischen dem Server und den Clients intakt? Diese und viele anderen Fragen können nicht beantwortet werden, bis ein standardisierter Netzwerklasttest für industriellen OPC UA-Geräten definiert ist.

Ähnlich, wie bei PROFINET, sollte das Thema der Netzlastrobustheit ein fester Bestandteil der Gerätezertifizierung von OPC UA-basierten Industrielösungen werden.

6 Referenzen:

[1] Heymann, Sascha; Stojanovic, Ljiljana; Watson, Kym; Seungwook, Nam; Song, Byunghun; Gschossmann, Hans; Schriegel, Sebastian; Jasperneite, Jürgen: Cloud-based Plug and Work architecture of IIC Testbed Smart Factory Web. In: IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA). Torino, Italy, September 2018

[2] M.Wollschläger, T.Sauter and J. Jasperneite, „The future of industrial communication“ in IEEE Industrial Electronics magazine. IEEE, Mar 2017

[3] „Industrie 4.0 Communication Guideline Based on OPC UA“ VDMA, Fraunhofer IOSB-INA 2017

[4] TSN Standards: IEEE 802.1Qbv : Enhancement for scheduled traffic , IEEE 802.1Qbu: Frame preemtion, IEEE 802.1AS-Rev/D2.0 : Timing and synchronization for time sensitive applications, IEEE 802.1Qci : Per-Stream Filtering and Policing

[5] I. Hübner: OPC UA TSN: Einheitlicher Kommunikationsstandard? In: Digital Factory Journal H. 1/2017 s.48. http://digital-factory-journal.de/uploads/tx_bcpageflip/DFJ_1_2017_Interaktiv.pdf

[6] Schriegel, Sebastian; Kobzan, Thomas; Jasperneite, Jürgen: Investigation on a Distributed SDN Control Plane Architecture for Heterogeneous Time Sensitive Networks. In: 14th IEEE International Workshop on Factory Communication Systems (WFCS) Imperia (Italy), Jun 2018

[7] Softing Industrial Automation GmbH (2018). Implementing Deterministic OPC UA Commmunication.

[8] OPC Foundation. OPC Unified Architecture Specification Part 14: PubSub Release 1.04, February 06, 2018

[9] OPC Unified Architecture – Part 7 : Profiles. IEC 62541-7, 2015

[10] J. Imtiaz and J. Jasperneite, "Scalability of OPC-UA down to the chip level enables “Internet of Things”," 2013 11th IEEE International Conference on Industrial Informatics (INDIN), Bochum, 2013, pp. 500-505

[11] OPC Unified Architecture – Part 2 : Security Model. IEC 62541-2, 2015

[12] Bundesamt für Sicherheit in der Informationstechnik „Sicherheitsanalyse OPC UA“ - Date April 2016. https://opcfoundation.org/wp-content/uploads/2016/04/Sicherheitsanalyse_OPC_UA_BSI_2016_v10-OPC-F-Responses.pdf
[13] OPC Foundation. How to certify. https://opcfoundation.org/certification/how-to-certify/
[14] „PROFINET IO Security Level 1 Guideline for PROFINET“ Version 1.2.1.1 – Date February 2017
[15] „Test Specification PROFINET IO Security Level 1 Technical Specification for PROFINET“ Version 1.1.6 – Date March 2017